Flashcards

File: Trees-annotated

Date: 30-07-2025

Total Cards: 10

Card 1/10

Q: What is the fundamental difference between a computer 'solving' a problem and 'executing' a solution, as stated by Laurent Gasser?

A: Computers don't inherently understand or devise solutions. They merely follow pre-programmed instructions (algorithms) to execute a solution that has already been defined by a human.

Card 2/10

Q: Explain the relationship between 'node', 'parent', and 'children' in the context of a tree data structure.

A: A 'node' is a basic unit of a tree. A 'parent' node is a node that has one or more nodes connected below it. These connected nodes are called 'children' of the parent node. A node can be both a parent and a child.

Card 3/10

Q: Differentiate between the 'degree of a node' and the 'degree of a tree'.

A: The 'degree of a node' is the number of children that node has. The 'degree of a tree' is the maximum degree of any node in the tree.

Card 4/10

Q: What distinguishes an 'internal node' from an 'external node' (leaf node) in a tree?

A: An 'internal node' (or non-terminal node) has at least one child. An 'external node' (or leaf node) has no children.

Card 5/10

Q: Describe the key differences between a 'strictly binary tree', a 'complete binary tree', and a 'full binary tree'.

A: A 'strictly binary tree' has each node with either 0 or 2 children. A 'complete binary tree' is filled from left to right on each level, with the last level possibly incomplete. A 'full binary tree' is a binary tree where every node has either 0 or 2 children, and all leaf nodes are at the same level.

Card 6/10

Q: Explain the purpose of an 'expression tree' and how it relates to prefix, infix, and postfix notations.

A: An 'expression tree' visually represents an arithmetic or logical expression. The placement of operators and operands in the tree dictates the order of operations. Traversing the tree in different ways (preorder, inorder, postorder) yields the prefix, infix, and postfix notations of the expression, respectively.

Card 7/10

Q: What is the significance of 'height' or 'depth' in the context of a tree, and how does it relate to the concept of 'level'?

A: The 'height' (or depth) of a tree is the maximum level of any node in the tree, starting from the root (usually level 0 or 1). 'Level' refers to the distance of a node from the root node.

Card 8/10

Q: Describe the difference between 'Depth First Search' and 'Breadth First Search' tree traversal methods.

A: 'Depth First Search' (DFS) explores as far as possible along each branch before backtracking. Common DFS traversals are Inorder, Preorder, and Postorder. 'Breadth First Search' (BFS) explores all the neighbor nodes at the present depth prior to moving on to the nodes at the next depth level.

Card 9/10

Q: What is a 'Binary Search Tree' (BST), and what property must it satisfy?

A: A 'Binary Search Tree' (BST) is a binary tree where for each node, the value of all nodes in its left subtree are less than the node's value, and the value of all nodes in its right subtree are greater than the node's value.

Card 10/10

Q: Explain the concept of a 'Heap' data structure and differentiate between a 'Max Heap' and a 'Min Heap'.

A: A 'Heap' is a specialized tree-based data structure that satisfies the heap property: In a 'Max Heap', the value of each node is greater than or equal to the value of its children. In a 'Min Heap', the value of each node is less than or equal to the value of its children.